Hierarchical Symbolic Sensitivity Computation with Applications to Large Amplifier Circuit Design*

Xiaopeng Li, Hui Xu and Guoyong Shi
School of Microelectronics, Shanghai Jiao Tong University
Shanghai 200240, China
shiguoyong@ic.sjtu.edu.cn

Andy Tai
Synopsys, Inc.
Mountain View, CA 94043, USA
Andy.Tai@synopsys.com

***Abstract*—Recently significant research progress has been made toward the exact computation of symbolic transfer functions for large analog networks containing over 40 MOS transistors. A successful application of such a computation method for analog design requires an efficient method for ac-sensitivity analysis. It is addressed in this paper that the sensitivity of a transfer function to device sizes (or RC values) can be computed efficiently based on a hierarchical framework. It is demonstrated via examples that the ac-sensitivity can be used for improving the amplifier design metrics such as phase margin or pole/zero placement, etc.**

***Index Terms*—ac-sensitivity, analog circuit design, binary decision diagram (BDD), device sizing, hierarchical symbolic analysis.**

I. Introduction

Symbolic circuit analysis is not popular because its capacity is considered limited. However, recently great progress has been made in extending the capacity of symbolic circuit simulator by applying advanced techniques such as Binary Decision Diagram (BDD) [1], [2]. Further extension of the analysis capacity is possible by incorporating well-developed hierarchical strategies, such as [3], [4] and recently [5]. Now exact symbolic analysis of op-amp circuits containing over 40 MOS transistors is possible.

A symbolic simulator that only generates a symbolic transfer function (even *exact*) is of limited use without imposing other useful functionalities. A useful extension is to add a sensitivity analysis feature [6]. Generally speaking, analytical sensitivity analysis would be much easier by using a symbolic approach than by a numerical approach where computation cost is a key concern. To avoid heavy computation involved in the adjoint network method, the work [7] introduced an approximate pole/zero analysis method by applying the moment matching technique. The recent work [8] addressed the symbolic ac-sensitivity to semiconductor device sizes (typically *width*) as a circuit characterization method. This paper continues along the same line to study a sensitivity computation method based on the hierarchical framework proposed in [5]. The application of symbolic sensitivity for circuit optimization, such as phase margin improvement and pole/zero placement is addressed.

*This research was supported in part by the National Natural Science Foundation of China (Grant No. 60876089) and by an SJTU-Synopsys Joint Research Grant (2010).

II. Background Review

A Binary Decision Diagram (BDD) [9] is a compact data structure for representation of symbolic expressions in Sum-Of-Product (SOP) form, which commonly appear in logic design and symbolic analog circuit analysis. Besides an algebraic application of BDD developed in the form of *DDD (Determinant Decision Diagram)*, another application of BDD for graph-based term enumeration called *GPDD (Graph-Pair Decision Diagram)* has been developed in [2], where a graph-pair reduction procedure is formulated in a BDD for *implicit* enumeration.

Since all BDD-based techniques are of exponential complexity (although lower than direct enumeration [10]), their capacity limits are reached for circuits containing about 20 MOS transistors. By exploiting structural circuit properties, such as common modular circuit structure, significant capacity improvement is still possible. The recent work [5] proposed to use one multi-root BDD to represent the symbolic stamp of a small-signal device model which is shared by all MOS transistors. The key idea was to incorporate the advantages of the two BDD-based symbolic techniques proposed in [1] and [2]. The computation method proposed in [5] is capable of analyzing *exactly* most practically encountered op-amp circuits.

III. Hierarchical AC-sensitivity Computation

The symbolic stamp computation in [5] made use of GPDD proposed in [2]. Based on the stamping theory in circuit simulation, the symbolic device stamps are assembled algebraically into a modified nodal analysis (MNA) matrix, which is then solved symbolically by DDD [1].

Suppose we have a connection of three circuit blocks shown in Fig. 1, where each block is a two-port and modeled by the following transadmittance matrix

$$\begin{pmatrix} i_1^\alpha \\ i_2^\alpha \end{pmatrix} = \begin{pmatrix} y_{11}^\alpha & y_{12}^\alpha \\ y_{21}^\alpha & y_{22}^\alpha \end{pmatrix} \begin{pmatrix} v_1^\alpha \\ v_2^\alpha \end{pmatrix}, \tag{1}$$

where α stands for the blocks A, B, and C.

Assembling the three two-port stamps according to their connection results in the following MNA matrix:

$$\begin{array}{c|cccc} & node_1 & node_2 & node_3 & node_4 \\ node_1 & y_{11}^A & & & y_{12}^A \\ node_2 & & y_{11}^B & & y_{12}^B \\ node_3 & & & y_{11}^C & y_{12}^C \\ node_4 & y_{21}^A & y_{21}^B & y_{21}^C & y_{22}^A + y_{22}^B + y_{22}^C \end{array} \tag{2}$$

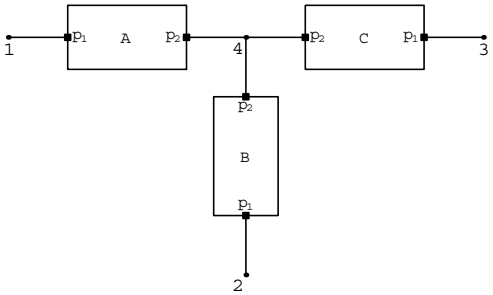


Fig. 1. A network with three blocks.

Circuit blocks other than two-ports can be treated similarly.

In the context of symbolic analysis, each entry of the MNA matrix is a summed expression of transadmittances, such as y_{ij}^A, etc., which are computed symbolically by a GPDD. A multi-root GPDD is able to represent the multiple transadmittances for one multi-port stamp in one shared BDD [5]. Shown in Fig. 2 is a multi-root BDD that represents the four entries of a 2×2 admittance matrix.

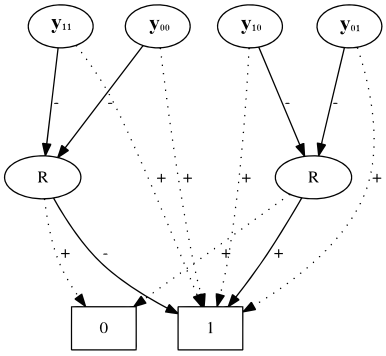


Fig. 2. A 2×2 admittance matrix represented by a four-root BDD.

Simply put, sensitivity analysis is just a mathematical *derivative* operation by the *Chain Rule*. In this sense, as long as the successive nesting of analytical functions are represented properly by a graphical data structure, a chained sequence of derivative operations can be implemented easily.

We suppose that the transfer function takes the following rational form

$$H(s) = \frac{N(s)}{D(s)}, \tag{3}$$

where both $N(s)$ and $D(s)$ are in *Sum-Of-Product* (SOP) forms and are represented by two BDDs linked hierarchically, with a DDD at the top-level.

The *ac-sensitivity* of $H(s)$ with respect to a parameter W is defined by the following normalized sensitivity equation,

$$\mathrm{Sens}(H(s), W) := \frac{W}{H(s)} \cdot \frac{\partial H(s)}{\partial W}. \tag{4}$$

Written in terms of $N(s)$ and $D(s)$, the above expression becomes

$$\mathrm{Sens}(H(s), W) = W \left(\frac{1}{N(s)} \frac{\partial N(s)}{\partial W} - \frac{1}{D(s)} \frac{\partial D(s)}{\partial W} \right). \tag{5}$$

Therefore, the ac-sensitivity is easy to obtain as long as the two derivatives $\frac{\partial N(s)}{\partial W}$ and $\frac{\partial D(s)}{\partial W}$ can be computed efficiently.

In the representation by DDD, the numerator $N(s)$ and denominator $D(s)$ can be accessed from the two child vertices of the DDD root. The derivatives are taken in the flow briefly described next. As part of the symbolic simulator construction, all devices appearing in a circuit are managed by their connection in circuit and model information. A MOSFET small-signal model is treated as a *model* (or a *subcircuit*), shown in Fig. 3, depending on the implementation. The subcircuit object knows where to access the associated entries in the MNA matrix and the associated DDD vertices. The subcircuit object also has a link to the GPDD data structure for fetching the transadmittance values and their derivatives with respect to the device size parameter W.

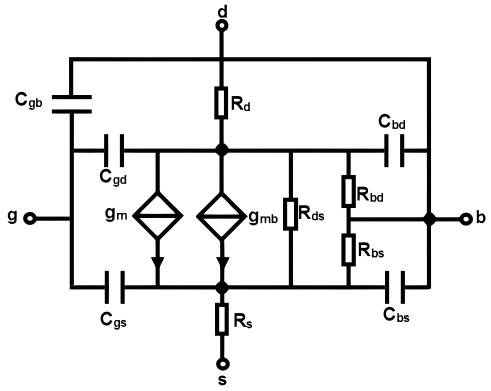


Fig. 3. MOS level 3 small signal model [11].

For example, in the saturation region, the small-signal parameters can be approximated by the following equations:

$$g_m = \sqrt{2k' \frac{W}{L} I_D}, \tag{6a}$$

$$R_{ds} = \frac{1}{\lambda I_D}, \tag{6b}$$

$$g_{mb} = \chi g_m, \tag{6c}$$

$$C_{sb} = C_{db} = C_j L_s W + C_{jsw}(2L_s + W), \tag{6d}$$

$$C_{gs} = \frac{2}{3} C_{ox} W L + C_o W, \tag{6e}$$

$$C_{gd} = C_o W. \tag{6f}$$

The derivative of the above quantities with respect to W are calculated as follows:

$$\frac{\partial g_m}{\partial W} = \frac{g_m}{2W}, \tag{7a}$$

$$\frac{\partial g_{mb}}{\partial W} = \frac{g_{mb}}{2W}, \tag{7b}$$

$$\frac{\partial C_{sb,db,gs,gd}}{\partial W} = \frac{C_{sb,db,gs,gd}}{W}, \tag{7c}$$

where it is assumed that the term $2L_s C_{jsw}$ is relatively small comparing to other terms in the expression for C_{sb} and C_{db}.

The BDD-based derivative evaluation is fairly simple and proceeds similarly for both DDD and GPDD. The

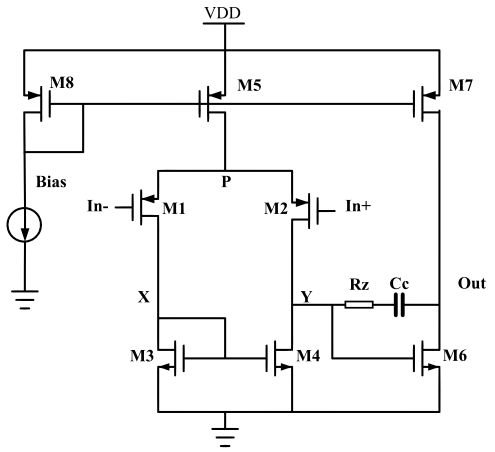


Fig. 4. Op-amp 1: A two stage amplifier with RC compensation.

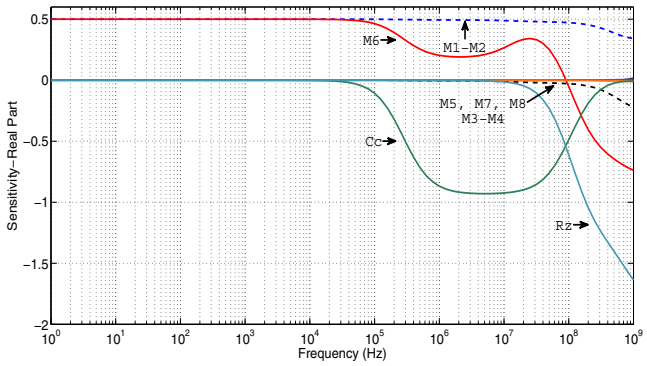


Fig. 5. Plot of $\mathrm{Sens}(|H(s)|, W_k)$.

basic vertex structure of BDD is a triple denoted by $(D.top,\ D.left,\ D.right)$, where the left-vertex $D.left$ and the right-vertex $D.right$ are connected from the top-vertex $D.top$ by a *solid* arrow (indicating a *multiplication*) and a *dashed* arrow (indicating an *addition*), respectively [1], [2].

With the above notation, the value of a top-vertex is calculated by the expression

$$Val(D) = D.top * Val(D.left) + Val(D.right), \qquad (8)$$

where the $Val(x)$ operator does the evaluation of the vertex x and $D.top$ stands for the *signed* symbol value at the top-vertex.

Applying the derivative operator $\frac{\partial}{\partial W}$ on the both sides of (8) results in

$$\frac{\partial D}{\partial W} = \frac{\partial (D.top)}{\partial W} * D.left + D.top * \frac{\partial (D.left)}{\partial W} + \frac{\partial (D.right)}{\partial W}. \qquad (9)$$

This key equation for sensitivity computation can be implemented by recursion.

IV. Experimental Results

A symbolic simulator with the feature of sensitivity analysis has been implemented in C++. The performance test results were collected from an AMD Athlon64 $2.20GHz$ processor with $2GB$ memory.

A simple two stage amplifier with RC compensation is given in Fig. 4. Initially, the transistors were sized at random just to make sure that all of them were biased in the saturation region without considering any specific design goals.

The following formulas tell us that the sensitivity of the magnitude $|H(s)|$ or phase $\angle H(s)$ with respect to a parameter p can be derived from the sensitivity of the transfer function $\mathrm{Sens}(H(s), p)$ [6].

$$\mathrm{Sens}(|H(s)|, p) = \mathrm{Re}\left\{\mathrm{Sens}(H(s), p)\right\}; \qquad (10)$$

$$\mathrm{Sens}(\angle H(s), p) = \frac{1}{\angle H(s)} \mathrm{Im}\left\{\mathrm{Sens}(H(s), p)\right\}. \qquad (11)$$

The magnitude sensitivities to the selected device sizes are plotted in Figs. 5 (similar plots for the phase sensitivities are omitted), from which we can see the relative levels of sensitivity and in addition the upward or downward ramps that can be shown to be associated with the poles or zeros.

The following observations can be made from the sensitivity plot in Fig. 5:

- Those biasing or active load transistors have low sensitivity to their device sizes.
- Those devices having rising/falling ramps in their magnitude sensitivity curves are pole-zero sensitive device, whose sizes can be adjusted accordingly to place the poles/zeros properly.

To demonstrate the use of the sensitivity information beyond the plots for visualization, we attempted to optimize the nulling resistor R_z so that the phase margin would meet a requirement.

After the initial sizing, the phase margin of the amplifier was about 25 degrees by HSPICE simulation. The reason was that the zero was apart from the second dominant pole (see Fig. 6(a)). We would like to adjust the phase margin to 60 degrees (a typical design requirement). From the plot in Fig. 5 we see that changing R_z does not affect the dominant pole at all, but it apparently affects the first zero. Hence, we attempted to increase the phase margin by adjusting the nulling resistor R_z using its phase sensitivity.

The R_z iteration is given by the following formula

$$R_z^{(n+1)} = R_z^{(n)} + \frac{\Delta(\angle H(s)) \cdot R_z^{(n)}}{\angle H(s) \cdot \mathrm{Sens}(\angle H(s), R_z^{(n)})}, \qquad (12)$$

where the phase sensitivity $\mathrm{Sens}(\angle H(s), R_z^{(n)})$ at the unity-gain frequency is used at each iteration.

TABLE I
Sensitivity-based resistor iteration.

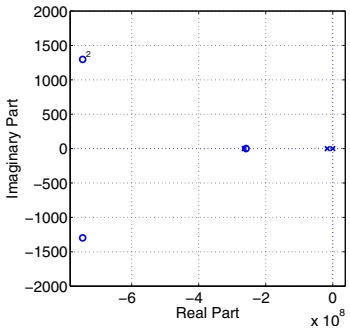
Iter. no.	$Rz(k\Omega)$	$\angle H(s)(deg)$	$Sens(\angle H(s), R_z)$	$\Delta R_z\ (\Omega)$
1	1	-155.6	-0.0661	$3,460$
2	4.46	-125	-0.2431	739
3	5.20	-120.5	-0.2347	95.5
4	5.29	-120		

After four iterations (Table I), the phase margin is seen very close to 60 degrees. We see from the plot of Fig. 6(b) that the

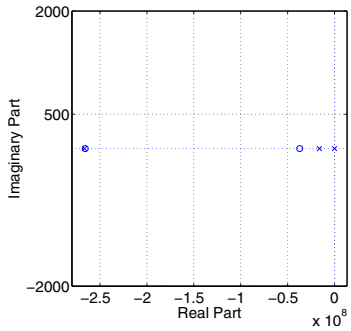
TABLE II
PERFORMANCE OF HIERARCHICAL SENSITIVITY ANALYSIS

Circuit	Opamp 1	Opamp 2	Opamp 3
No. transistors	8	24	44
GPDD build time (sec)	0.1	0.1	0.1
DDD build time (sec)	0.01	0.18	0.3
Evaluation time (sec)	0.28	23.23	57.02
Total time (sec)	0.38	23.51	57.42
Memory (MB)	56	101	153

new zero has moved closer toward the second dominant pole after optimization. Comparing to the work of [7] where the approximate pole-zero sensitivity was computed with quite some effort by moment matching, the sensitivity calculation method presented here is much simpler for parametric optimization.



(a) Pole and zero location before R_z was adjusted.



(b) Pole and zero location after R_z iteration.

Fig. 6. Dominant poles and zeros by HSPICE.

The hierarchical sensitivity computation method has been applied to solve larger op-amp circuits as well in our experiment, such as the one (op-amp 2) shown in Fig. 7 which contains 24 MOS transistors and another op-amp (op-amp 3 not shown) containing 44 transistors. The performance of our symbolic simulator is summarized in Table II, in which the time and memory for calculating the sensitivities to all transistors were obtained at runtime and the evaluation time was measured for analyzing 100 frequency points.

For the purpose of illustration, the sensitivity plot of op-amp 2 in Fig. 7 with respect to some selected transistors is given in Fig. 8 (similar plot for the phase sensitivity is omitted), where both the sensitivity levels and the pole/zero locations (where the curves ramp up or down) can be seen clearly.

V. CONCLUSION

This paper presents briefly a hierarchical sensitivity analysis method based on a recently developed hierarchical computation scheme that makes use of two layers of BDDs. A design example is presented to demonstrate the advantages of using such an analysis tool. More details on the technical aspects mentioned in the paper will be discussed in other publications.

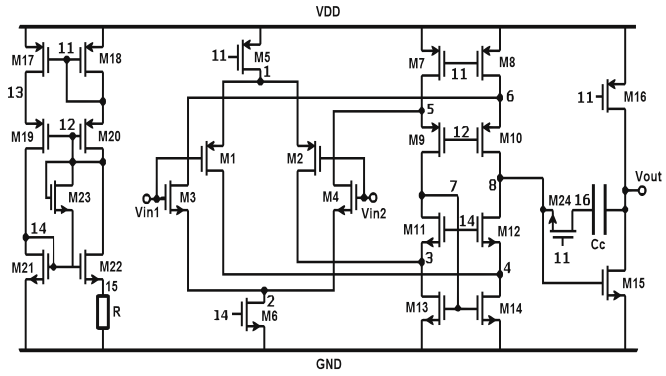


Fig. 7. Op-amp 2: A rail-to-rail folded-cascode CMOS amplifier

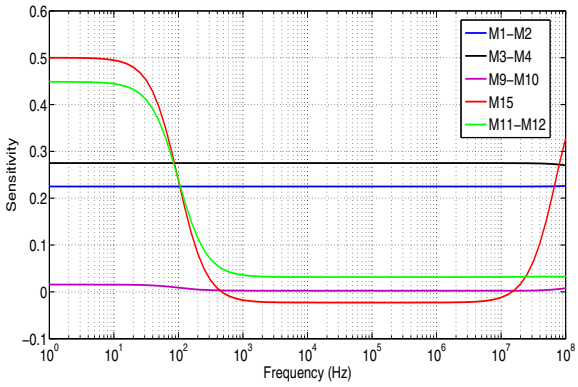


Fig. 8. Magnitude Sensitivity

REFERENCES

[1] C. J. R. Shi and X. D. Tan, "Canonical symbolic analysis of large analog circuits with determinant decision diagrams," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, vol. 19, no. 1, pp. 1–18, January 2000.

[2] G. Shi, W. Chen, and C. J. R. Shi, "A graph reduction approach to symbolic circuit analysis," in *Proc. Asia South-Pacific Design Automation Conference (ASPDAC)*, Yokohama, Japan, Jan. 2007, pp. 197–202.

[3] X. D. Tan and C. J. R. Shi, "Hierarchical symbolic analysis of analog integrated circuits via determinant decision diagrams," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, vol. 19, no. 4, pp. 401–412, April 2000.

[4] S. X. D. Tan, W. Guo, and Z. Qi, "Hierarchical approach to exact symbolic analysis of large analog circuits," in *Proc. Design Automation Conference*, 2004, pp. 860–863.

[5] H. Xu, G. Shi, and X. Li, "Hierarchical exact symbolic analysis of large analog integrated circuits by symbolic stamps," in *Prof. Asia South-Pacific Design Automation Conference (ASPDAC)*, Yokohama, Japan, Jan. 2011, accepted for publication.

[6] G. Shi and X. Meng, "Variational analog integrated circuit design by symbolic sensitivity analysis," in *Proc. International Symposium on Circuits and Systems (ISCAS)*, Taiwan, China, May 2009, pp. 3002–3005.

[7] J. Y. Lee, X. Huang, and R. A. Rohrer, "Pole and zero sensitivity calculation in asymptotic waveform evaluation," *IEEE Trans. on Computer-Aided Design*, vol. 11, no. 5, pp. 586–597, May 1992.

[8] D. Ma, G. Shi, and A. Lee, "A design platform for analog device size sensitivity analysis and visualization," in *Proc. Asia Pacific Conference on Circuits and Systems (APCCAS)*, Malaysia, Dec. 2010, pp. 48–51.

[9] R. E. Bryant, "Graph-based algorithms for boolean function manipulation," *IEEE Trans. on Computers*, vol. C-35, no. 8, pp. 677–691, 1986.

[10] G. Shi, "Computational complexity analysis of determinant decision diagram," *IEEE Trans. on Circuits and Systems - II: Express Briefs*, vol. 57, no. 10, pp. 828–832, 2010.

[11] A. Vladimirescu and S. Liu, "The simulation of MOS integrated circuits using SPICE2," EECS Department, University of California, Berkeley, Tech. Rep. UCB/ERL M80/7, 1980. [Online]. Available: http://www.eecs.berkeley.edu/Pubs/TechRpts/1980/9610.html